JUL 26 2004 JC37 TRADEMARK OFFICE



RECEIVED
JUL 2 9 2004
Technology Center 2100

**IN THE CLAIMS:**

1. (Currently Amended): A data authentication system comprising:

A. an integrity check processor that

i. selects one or more integrity functions from a set of functions, and

ii. manipulates m selected data bytes from each of one or more data packets in accordance with the selected integrity check functions to produce one or more integrity checks that correspond to the one or more data packets; and

B. an integrity block processor that so encrypts the one or more integrity checks produced by the integrity check processor as to permit their decryption only with a non-public key and produces an integrity block that is used to authenticate the data packets.

2. (Original): The data authentication system of claim 1 wherein the integrity check processor includes in the integrity check an indication of which integrity function to select.

3. (Original): The data authentication system of claim 2 wherein the indication is a function identifier.

4. (Original): The data authentication system of claim 2 wherein the indication is an offset value for a pseudorandom sequence known to a sender and an intended recipient.

5. (Original): The data authentication system of claim 4 wherein the pseudorandom sequence is generated using a seed value known by the sender and the intended recipient.

6. (Original): The data authentication system of claim 1 wherein the integrity check processor uses information in the one or more data packets as one or more offset values for a pseudorandom sequence known to a sender and an intended recipient.

7. (Original): The data authentication system of claim 6 wherein the pseudorandom sequence is generated using a seed value known by the sender and the intended recipient.

8. (Original): The data authentication processor of claim 2 wherein the integrity check processor selects more than one integrity function for a given data packet and includes in the integrity check information that identifies a list of the selected functions and a corresponding list of the results of the manipulations.

9. (Original): The data authentication system of claim 1 wherein the integrity block processor encrypts the integrity checks in accordance with a secret key that is shared by intended recipients of the data packets.

10. (Original): The data authentication system of claim 1 wherein the integrity check processor selects the m data bytes at random from a first data packet, and for any remaining data packets selects data bytes that are offset from the data bytes selected from the first data packet.

11. (Original): The data authentication system of claim 1 wherein the integrity block processor encrypts into the integrity block information that identifies the data bytes selected from each of the data packets.

12. (Original): The data authentication system of claim 11 wherein the information includes data byte interval and offset values.

13. (Original): The data authentication system of claim 1 wherein the integrity check processor includes in the integrity checks one or more sequence numbers that are associated with the data packets.

14. (Original): The data authentication system of claim 1 wherein the integrity block processor assembles the plurality of integrity checks in an order that differs from the order of the data packets and encrypts into the integrity block information that associates the integrity checks with the appropriate data packets.

15. (Original): The data authentication system of claim 14 wherein the integrity block processor encrypts into the integrity block a list of sequence numbers that corresponds to the order of the integrity checks within the integrity block.

16. (Original): The data authentication system of claim 1 wherein the integrity check processor produces digital signatures for one or more of the data packets and includes the digital signatures in the respective data packets.

17. (Original): The data authentication system of claim 1 wherein the integrity block processor produces a digital signature for the integrity block and includes the digital signature in the integrity block.

18. (Original): The data authentication system of claim 1 wherein the selected integrity check function concatenates the selected data bytes from a given data packet to produce the associated integrity check.

19. (Original): The data authentication system of claim 1 further including a chaff processor for producing for transmission extraneous packets that are associated with and do not pass one or more of the integrity checks, the chaff processor including the extraneous packets in a transmission that includes the data packets.

20. (Original): The data authentication system of claim 1 wherein the integrity block processor encrypts into the integrity block executable code that performs the selected integrity check function.

21. (Original): The data authentication system of claim 20 wherein the integrity block processor signs the executable code with a digital signature.

22. (Currently Amended): A communications network comprising:

A. one or more sending stations for sending data packets;

B. one or more recipient stations for receiving the data packets sent by the sending stations; and

C. an authentication system that includes

i. an integrity block processor for:

a. ~~for~~ selecting one or more integrity functions from a set of integrity functions,

b. manipulating one or more selected data bytes from a given data packet in accordance with the one or more selected integrity check functions to produce the corresponding integrity check, and

c. so encrypting the one or more integrity checks that are associated with one or more data packets as to permit their decryption only with a non-public key, ~~to produce~~ producing therefrom an integrity block, and including the integrity block in a transmission to the recipient stations, and

ii. authentication means for decrypting a received integrity block to reproduce the one or more integrity checks and using information contained in the reproduced integrity checks to select one or more

integrity check functions and one or more data bytes to use to determine if data in the associated one or more data packets have been altered.

23. (Original): The communications network of claim 22 wherein the authentication means selects the one or more integrity check functions for use in authenticating the data packets based on identifying information in the associated integrity check.

24. (Original): The communications network of claim 22 wherein the authentication means uses information in the integrity check or in the associated data packet as an offset value into a pseudo random sequence known to the sender and an intended recipient and uses the next n bits of the sequence to identify the selected integrity check.

25. (Original): The communications network of claim 22 wherein the authentication means uses the one or more integrity checks, the integrity check functions identified therein and the selected data bytes from the one or more data packets to determine if the data packets have been altered.

26. (Original): The communications network of claim 22 wherein the integrity block processor is included in each of the one or more sending stations and the authentication means is included in each of the one or more recipient stations.

27. (Original): The communications network of claim 22 wherein the integrity block processor encrypts the integrity checks and the authentication means decrypts the integrity blocks in accordance with one or more secret keys that are shared by the sending stations and the intended recipient stations.

28. (Original): The communications network of claim 22 wherein the integrity block processor selects one or more data bytes at random from a first data packet and selects

from the remaining data packets data bytes that are offset from the data bytes selected from the first data packet based on the information contained in the associated integrity checks.

29. (Original): The communications network of claim 22 wherein the integrity block processor encrypts into an integrity block the information that identifies the integrity check function.

30. (Original): The communications network of claim 22 wherein the integrity block processor encrypts into an integrity block the information that identifies the data bytes selected for each of the one or more data packets by the integrity block processor.

31. (Original): The communications network of claim 30 wherein the information includes data byte interval and offset values.

32. (Original): The communications network of claim 22 wherein the integrity block processor further includes in the integrity block sequence numbers that correspond to the associated data packets.

33. (Original): The communications network of claim 22 wherein the authentication means assembles the integrity checks in an order that differs from the order of the associated data packets and encrypts into the integrity block information that associates the integrity checks with the appropriate data packets.

34. (Original): The communications network of claim 33 wherein the authentication means further encrypts into the integrity block a list of data packet sequence numbers that corresponds to the order of the integrity checks within the integrity block.

35. (Original): The communications system of claim 22 wherein the authentication means further produces a digital signature for each data packet and includes the digital signature in the data packet.

36. (Original): The communications system of claim 22 wherein the authentication means concatenates selected data bytes from a given data packet to produce the associated integrity check.

37. (Original): The communications system of claim 22 wherein the authentication means encodes selected bytes from a given data packet to produce the associated integrity check.

38. (Original): The communications system of claim 22 further including a chaff processor that produces for transmission one or more extraneous packets that are associated with and do not pass one or more of the integrity checks, the chaff processor including the extraneous packets in a transmission with the associated data packets.

39. (Original): The communications system of claim 22 wherein the integrity block processor further includes in the integrity block executable code that performs an integrity check process.

40. (Original): The communications system of claim 39 wherein the integrity block processor includes in an integrity block a digital signature that corresponds to the executable code.

41. (Currently Amended): A method of authenticating data that is sent in data packets, the method including the steps of:

A. selecting one or more integrity functions from a set of integrity functions;

B. manipulating selected data bytes from a first data packet in accordance with one or more of the selected integrity functions to produce an integrity check;

C. so encrypting the integrity check as to produce an integrity block in which the integrity check can be decrypted only with a non-public key;

D. sending the integrity block to intended recipients.

42. (Original): The method of claim 41 further including the steps of:

E. decrypting a received integrity block to reproduce the integrity check;

F. selecting one or more integrity check functions from the set of functions;

G. using the reproduced integrity check and the selected integrity check functions to determine if the first data packet is authentic.

43. (Original): The method of claim 42 further including the steps of

H. manipulating data bytes from additional data packets in accordance with one or more of the selected integrity check functions to produce additional integrity checks;

I. encrypting the additional integrity checks into the integrity block;

J. decrypting the received integrity block to reproduce the additional integrity checks;

K. selecting one or more integrity check functions; and

L. using the reproduced additional integrity checks and the selected integrity check functions to determine if respective additional data packets are authentic.

44. (Original): The method of claim 41 wherein the step of selecting the integrity functions includes providing associated identifiers as part of the integrity check.

45. (Original): The method of claim 41 wherein the step of selecting the integrity functions includes

i. using information in the data packet as an offset value into a pseudorandom sequence, and

ii. using the next n bits of the sequence as the integrity function identifier.

46. (Original): The method of claim 43 further including in the step of encrypting the integrity checks, performing the encryption in accordance with a secret key that is available to the recipients.

47. (Original): The method of claim 46 further including in the step of decrypting the integrity block, decrypting the block in accordance with the secret key.

48. (Original): The method of claim 43 wherein the step of manipulating data bytes selects the data bytes at random from the first data packet and selects from the additional data packets data bytes that are offset from the data bytes selected from the first data packet.

49. (Original): The method of claim 43 wherein the step of encrypting the integrity checks further includes encrypting into the integrity block information that identifies the data bytes selected from the data packets.

50. (Original): The method of claim 43 further including in the step of encrypting the integrity checks the step of encrypting into the integrity block data byte interval and offset values.

51. (Original): The method of claim 43 wherein the step of manipulating the data bytes to produce the integrity checks further includes the step of including in the integrity checks sequence numbers that correspond to the associated data packets.

52. (Original): The method of claim 43 wherein the step of encrypting the integrity checks includes assembling the integrity checks in an order that differs from the order of the associated data packets.

53. (Original): The method of claim 52 wherein the encrypting step further includes the step of encrypting into the integrity block a list of sequence numbers that corresponds to the order of the integrity checks.

54. (Original): The method of claim 43 further including the step of producing a digital signature for each data packet and including the digital signature in the data packet.

55. (Original): The method of claim 42 further including the step of producing a digital signature for the integrity block and including the signature in the block.

56. (Original): The method of claim 43 wherein the step of manipulating the selective data bytes includes concatenating the selected data bytes from a given data packet to produce the associated integrity check.

57. (Original): The method of claim 43 wherein the step of manipulating the selected data bytes includes encoding the selected bytes from a given data packet to produce the associated integrity check.

58. (Original): The method of claim 42 further including the step of including in a transmission extraneous packets that are associated with and do not pass one or more of the integrity checks.

59. (Original): The method of claim 42 wherein the step of encrypting the integrity checks further includes encrypting into the integrity block executable code that performs an integrity check process.

60. (Original): The method of claim 59 wherein the encrypting step further includes encrypting into the integrity block a digital signature associated with the code.

61. (Currently Amended): A data authentication system comprising:

A. an integrity block processor that receives a plurality of data packets and an associated integrity block, the integrity block processor manipulating the integrity block to produce a plurality of integrity checks that correspond to the data packets, and

B. an integrity check processor that employs a non-public key to decrypt the integrity block and thereby produce the plurality of integrity checks and that uses the integrity checks, integrity check functions selected from a set of functions and selected data bytes from the data packets to determine if any of the data packets have been altered.

62. (Original): The authentication system of claim 61 wherein the integrity block processor further produces from the integrity block information to determine which data bytes to select from the data packets.

63. (Original): The authentication system of claim 61 wherein the integrity block processor produces from the integrity block information to select which integrity check functions to use to manipulate the selected data packets.

64. (Original): The authentication system of claim 63 wherein the information determines which function or functions to use for each data packet.

65. (Canceled)

66. (Currently Amended): The authentication system of claim ~~65~~ 61 wherein the integrity block processor uses a shared secret key to decrypt the integrity block.

67. (Currently Amended): The authentication system of claim ~~65~~ 61 wherein the integrity block processor decrypts the integrity block to provide to the integrity check processor executable code to use to manipulate the selected data bytes.

68. (Currently Amended): The authentication system of claim 62 wherein ~~the integrity block processor decrypts the integrity block to produce the integrity checks and~~ the integrity check processor uses information in the integrity checks to determine which data bytes to select from the one or more data packets.

69. (Original): The authentication system of claim 63 wherein the integrity check processor uses a digital signature included in the integrity block to authenticate the integrity block.

70. (Original): The authentication system of claim 61 wherein the integrity check processor uses one or more digital signatures included in the one or more data packets to further authenticate the data packets.

71. (Currently Amended): A system for authenticating one or more data packets, the system comprising:

A. means for configuring at least one sending station with an authentication process adapted to produce an encrypted integrity block from one or more integrity checks associated with one or more data packets and one or more integrity functions selected from a set of integrity functions, which integrity block is so encrypted as to permit its decryption only with a non-public key;

B. means for configuring at least one receiving station with an authentication process adapted to employ a non-public key to decrypt a received integrity block into one or more integrity checks and authenticate the associated one or more data packets using the one or more integrity checks and associated selected integrity functions.

72. (Original): The system of claim 71 wherein the receiving station selects, based on information contained in the integrity block, the one or more integrity functions from the set of functions and one or more selected data bytes from each of the one or more packets to use in the authentication process.

73. (Original): The system of claim 71 wherein the means for configuring at least one sending station includes a computer readable medium containing executable program instructions.

74. (Original): The system of claim 71 wherein the means for configuring at least one receiving station includes a computer readable medium containing executable code.

75. (Original): The system of claim 71 further including means for configuring the sending station to transmit extraneous data packets that are associated with the integrity block but do not pass authentication.

76. (Currently Amended): A computer data signal embodied in a carrier wave and representing sequences of instructions for authenticating data packets, the instructions comprising instructions for:

configuring at least one sending station to produce an encrypted integrity block for a plurality of data packets using one or more integrity check functions selected from a set of integrity check functions, which integrity block is so encrypted as to permit its decryption only with a non-public key; and

at the configured sending station selecting one or more data bytes from each data packet and producing an associated integrity check that is used with the integrity checks for the other data packets to produce the encrypted integrity block.

77. (Original): The computer data signal of claim 76 wherein the selection of data bytes from a first data packet is random and the data bytes selected from remaining data packets are offset from the data bytes selected from first data packet.

78. (Original): The computer data signal of claim 76 wherein the integrity block is encrypted in accordance with a shared secret key.

79. (Original): The computer data signal of claim 76 wherein the one or more integrity checks are produced by concatenating selected data bytes from respective data packets.

80. (Original): The computer data signal of claim 76 wherein the one or more integrity checks are produced by encoding selected data bytes from respective data packets.

81. (Original): The data signal of claim 76 further comprising instructions for

configuring at least one receiving station to decrypt the encrypted integrity block to reproduce the one or more integrity checks; and

at the configured receiving station using the one or more integrity checks to authenticate the one or more data packets.

82. (Original): The computer data signal of claim 81 wherein the one or more integrity checks are associated with the appropriate one or more data packets prior to authentication.

83. (Original): The computer data signal of claim 76 further including configuring the sending station to transmit one or more extraneous data packets that are associated with the integrity block but do not pass authentication tests.

84. (Currently Amended): A ~~data authentication system in which sequences of instructions for authenticating data packets are stored on a~~machine-readable medium~~,~~ that store~~the~~ instructions ~~comprising instructions~~ for configuring at least one sending station to:

~~configuring at least one sending station to~~ produce ~~an encrypted integrity block~~ for one or more data packets an integrity block so encrypted as to permit its decryption only with a non-public key; and

~~at the configured sending station selecting~~select one or more data bytes from the one or more data packets ~~and~~

~~producing~~ produce one or more integrity checks using integrity functions that are selected from a set of functions, and encrypting the results and information that identifies the selected functions for each packet to produce the encrypted integrity block.